

Programação 1 - Prof. André Grégio
Maisa Girardi - GRR20204081

1. Escreva um programa em C que:

- a. Receba um inteiro n do usuário;
- b. Crie um vetor de tamanho n, desde que n seja igual ou menor do que 1024;
- c. Preencha cada posição do vetor com o valor dos índices do intervalo;
- d. **LEMBRETE:** o tamanho máximo do vetor deve ser **ESTÁTICO**!

RESPOSTA:

```
#include <stdio.h>

int main () {
    int i, n, v[n];

    scanf ("%d", &n);

    if (n <= 1024) {
        for (i = 0; i < n; i++) {
            v[i] = i;
            printf ("%d\n", v[i]);
        }
    }
    return 0;
}
```

Entrada: 5

Saída: 0 1 2 3 4

```
> clang-7 -pthread -lm -o main main.c
> ./main
5
0
1
2
3
4
> 
```

2. Com base no programa do item 1, escreva um programa em C que:

a. No item “c” do item anterior, em vez de preencher cada posição do vetor com o valor dos índices do intervalo, preencha o vetor em modo decrescente (ex.: se $n == 10$, os índices do vetor vão de 0 até 9; a posição indexada por 0 deve receber o valor 9, posição 1 recebe 8 e assim por diante até que o índice 9 receba valor 0).

RESPOSTA:

```
#include <stdio.h>

int main () {

    int n, i, aux, v[1024];

    scanf ("%d", &n);
    aux = n - 1;
    if (n <= 1024) {
        for (int i = 0; i < n; i++){
            v[i] = aux;
            printf ("%d\n", v[i]);
            aux = aux - 1;
        }
    }
    return 0;
}
```

Entrada: 10

Saída: 9 8 7 6 5 4 3 2 1 0

```
❖ clang-7 -pthread -lm -o main main.c
❖ ./main
10
9
8
7
6
5
4
3
2
1
0
❖
```

3. Com base no programa do item 1, escreva um programa em C que:

- a. No item “c”, receba do usuário cada valor a ser indexado no intervalo [0, n);
- b. Imprima o vetor recebido em ordem.

RESPOSTA:

```
#include <stdio.h>

int main () {

    int n, i, v[1024];

    scanf ("%d", &n);
    if (n <= 1024) {
        for (int i = 0; i < n; i++){
            scanf ("%d", &v[i]);
        }
        printf ("%d", v[n]);
    }

    return 0;
}
```

Entrada: 3 4 2 6

Saída: 4 2 6

```
❯ clang-7 -pthread -lm -o main main.c
❯ ./main
3
4
2
6
0❯
```

4. Com base no programa do item 3, escreva outro programa em C que:

- a. Após ter um vetor de tamanho n passado pelo usuário e preenchido com os valores de índices em ordem decrescente, imprima o vetor do fim para o começo, isto é, apresente o vetor na tela da posição n-1 até a posição 0.

RESPOSTA:

```
#include <stdio.h>

int main () {

    int n, i, v[1024];
```

```

scanf ("%d", &n);
if (n <= 1024) {
    for (int i = 0; i < n; i++){
        scanf ("%d", &v[i]);
    }

    do {
        printf ("%d\n", v[n]);
        n = n - 1;
    } while (n >= 0);
}

return 0;
}

```

Entrada: 5 20 15 13 10 6

Saída: 6 10 13 15 20

```

➤ clang-7 -pthread -lm -o main main.c
➤ ./main
5
20
15
13
10
6
0
6
10
13
15
20

```

5. Escreva um programa em C para cada um dos itens abaixo:

a. Um vetor preenchido é copiado em ordem inversa em um outro vetor.

RESPOSTA a.:

```

#include <stdio.h>

int main () {

    int n, i, v[1024], v1[1024], aux;

    scanf ("%d", &n);
    if (n <= 1024) {

```

```

        for (int i = 0; i < n; i++){
            scanf ("%d", &v[i]);
        }
    }
    aux = n;
    for (int i = 0; i <= aux; aux--){
        v1[i] = v[aux];
        printf ("%d\n", v1[i]);
    }

    return 0;
}

```

Entrada: 4 1 2 3 4

Saída: 4 3 2 1

```

> clang-7 -pthread -lm -o main main.c
> ./main
4
1
2
3
4
0
4
3
2
1
>

```

b. Calcule a soma dos elementos de um vetor e a apresente na tela.

RESPOSTA b.:

```

#include <stdio.h>

int main () {

    int n, i, v[1024], soma;

    scanf ("%d", &n);
    if (n <= 1024) {
        for (int i = 0; i < n; i++){
            scanf ("%d", &v[i]);
        }
    }
}

```

```

soma = 0;
for (int i = 0; i < n; i++){
    soma = soma + v[i];
}

printf ("%d\n", soma);

return 0;
}

```

Entrada: 4 1 2 5 7

Saída: 15

```

❯ clang-7 -pthread -lm -o main main.c
❯ ./main
4
1
2
5
7
15
❯

```

c. Calcule a média dos elementos de um vetor e a apresente na tela.

RESPOSTA c.:

```

#include <stdio.h>

int main () {

    int n, i, v[1024], soma, media;

    scanf ("%d", &n);
    if (n <= 1024) {
        for (int i = 0; i < n; i++){
            scanf ("%d", &v[i]);
        }
    }

    soma = 0;
    for (int i = 0; i < n; i++){
        soma = soma + v[i];
    }
}

```

```

media = 0;
media = soma / (float) n;

printf ("%d", media);

return 0;
}

```

Entrada: 3 4 2 6

Saída: 4

```

> clang-7 -pthread -lm -o main main.c
> ./main
3
4
2
6
4>

```

d. Imprime o maior e o menor elemento deste vetor.

RESPOSTA d.:

```

#include <stdio.h>

int main () {
    int n, i, v[1024], maior, menor;

    scanf ("%d", &n);
    if (n <= 1024) {
        for (int i = 0; i < n; i++){
            scanf ("%d", &v[i]);
        }

        maior = v[0];
        for (i = 1; i < n; i++){
            if (v[i] > maior) {
                maior = v[i];
            }
        }

        menor = v[0];

```

```

        for (i = 1; i < n; i++){
            if (v[i] < menor) {
                menor = v[i];
            }
        }
    }
    printf ("Maior = %d\n", maior);
    printf ("Menor = %d\n", menor);

    return 0;
}

```

Entrada: 4 8 5 2 4

Saída: Maior = 8

Menor = 2

```

> clang-7 -pthread -lm -o main main.c
> ./main
4
8
5
2
4
Maior = 8
Menor = 2
>

```

6. Escreva um programa em C no qual, dados dois vetores de mesmo tamanho, multiplique cada posição de cada vetor e guarde o produto em um terceiro vetor.

RESPOSTA:

```

#include <stdio.h>

int main () {

    int n, i;

    scanf ("%d", &n);

    int v1[n], v2[n], v3[n];

    if (n <= 1024) {
        for (int i = 0; i < n; i++){
            scanf ("%d", &v1[i]);

```



```

    }

    for (int i = 0; i < n; i++){
        scanf ("%d", &v2[i]);
    }

    for (int i = 0; i < n; i++){
        v3[i] = v1[i] * v2[i];
        printf ("%d\n", v3[i]);
    }

    return 0;
}

```

Entrada: 3 1 2 3 5 3 4

Saída: 5 6 12

```

> clang-7 -pthread -lm -o main main.c
> ./main
3
1
2
3
5
3
4
5
6
12
>

```

7. Escreva um programa em C que conte quantos elementos distintos há em um vetor e apresente cada elemento e sua frequência como saída.

a. Exemplo, para $v[5] = \{1, 1, 1, 3, 3\}$, a saída deve ser:

Elemento 1 apareceu 3 vezes

Elemento 3 apareceu 2 vezes

RESPOSTA:

```

#include <stdio.h>

int main() {
    int n, i, aux, j;
    int cont = 0;

```

```

scanf ("%d", &n);
int v[n];

for (int i = 0; i < n; i++){
    scanf ("%d", &v[i]);
}
j=0;
while (j < n){
    aux = v[j];
    for (i = 0; i < n; i ++){
        if (aux == v[i]){
            cont++;
            j++;
        }
    }
    printf ("Elemento %i apareceu %d vezes\n", aux, cont);
    cont=0;
}
return 0;
}

```

Entrada: 5 1 1 1 3 3

Saída: Elemento 1 apareceu 3 vezes

Elemento 3 apareceu 2 vezes

```

❯ clang-7 -pthread -lm -o main main.c
❯ ./main
5
1
1
1
3
3
Elemento 1 apareceu 3 vezes
Elemento 3 apareceu 2 vezes
❯

```

8. Escreva um programa em C no qual, dados dois vetores do mesmo tamanho, concatenar-os em um terceiro vetor.

RESPOSTA:

```
#include <stdio.h>

void lerVetor (int n, int v[]) {
    for (int i = 0; i < n; i++){
        scanf ("%d", &v[i]);
    }
}

void imprimir (int tam, int v[]){
    for (int i = 0; i < tam; i++){
        printf ("%d", v[i]);
    }
}

int main () {
    int n, i;

    scanf ("%d", &n);
    int v1[n], v2[n];

    //concatenacao dos vetores v1 e v2
    int tam = n + n;
    int v3[tam];

    lerVetor (n, v1);
    lerVetor (n, v2);

    for (int i = 0; i < n; i++) {
        v3[i] = v1[i];
    }
    int aux = 0;
    for (int i = n; i < tam; i++){
        v3[i] = v2[aux];
        aux++;
    }
    imprimir (tam, v3);
    return 0;
}
```

Entrada: 3 1 2 3 3 4 6

Saída: 1 2 3 3 4 6

```
➤ clang-7 -pthread -lm -o main main.c
➤ ./main
3
1
2
3
3
4
6
123346➤
```

9. Escreva um programa em C que mostre, para um dado vetor:

a. Quantos e quais são os elementos repetidos neste vetor;

RESPOSTA:

```
#include <stdio.h>

int main() {
    int n, i, aux, j, num;
    int cont = 0;

    scanf ("%d", &n);
    int v[n];
    int v2[n];

    for (int i = 0; i < n; i++){
        scanf ("%d", &v[i]);
    }

    for (i = 0; i < n; i++){
        for (j = i+1; j < n; j++){
            if (v[i] == v[j]){
                cont++;
                num = v[i];
            }
        }
    }

    printf ("Quantidade = %d\n", cont);
    printf ("Qual numero: %d\n", num);
    return 0;
}
```

```
}
```

Entrada: 5 1 1 1 3 2

Saída: Quantidade = 3

Qual numero: 1

```
❯ clang-7 -pthread -lm -o main main.c
❯ ./main
5
1
1
1
3
2
Quantidade = 3
Qual numero: 1
❯
```

10. Escreva um programa em C que, dado um vetor preenchido, imprime a quantidade de elementos pares e ímpares.

RESPOSTA:

```
#include <stdio.h>

int main () {
    int n, i, j;
    int par, impar = 0;

    scanf ("%d", &n);
    int v[n];

    for (int i = 0; i < n; i++){
        scanf ("%d", &v[i]);
        if (v[i]%2 == 0) {
            par = par + 1;
        }
        else {
            impar = impar + 1;
        }
    }
    printf ("Pares = %d\n", par);
    printf ("Impares = %d\n", impar);
    return 0;
}
```

Entrada: 5 3 2 6 1 0
Saída: Pares = 3
Impares = 2

```
➤ clang-7 -pthread -lm -o main main.c  
➤ ./main  
5  
3  
2  
6  
1  
0  
Pares = 3  
Impares = 2  
➤
```